EFFECTIVE PREDICTION OF PARKINSON’S DISEASE

Adarsh G	(20Z204)
Elanthamil R	(20Z215)
Jeevan Krishna K V	(20Z220)
Nirmal M	(20Z267)
Ajay Deepak P M	(21Z431)

Dissertation submitted in partial fulfillment of the requirements for the degree of

BACHELOR OF ENGINEERING
BRANCH: COMPUTER SCIENCE AND ENGINEERING

APRIL - 2023

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
PSG COLLEGE OF TECHNOLOGY
(Autonomous Institution)
COIMBATORE – 641 004

PSG COLLEGE OF TECHNOLOGY

(Autonomous Institution)
COIMBATORE – 641 004

COMPUTER SCIENCE AND ENGINEERING

EFFECTIVE PREDICTION OF PARKINSON’S DISEASE

Bona fide record of work done by

Adarsh G	**(20Z204)**
Elanthamil R	**(20Z215)**
Jeevan Krishna K V	**(20Z220)**
Nirmal M	**(20Z267)**
Ajay Deepak P M	**(21Z431)**

Dissertation submitted in partial fulfillment of the requirements for the degree of

BACHELOR OF ENGINEERING

BRANCH: COMPUTER SCIENCE AND ENGINEERING

……………………………   ……………………………

Dr. G. Sudha Sadasivam
Head of the Department

Dr. Gopika Rani N
Faculty Incharge

Certified that the candidates were examined in the viva-voce examination held on 12-04-2023

……………………………   ……………………………

(Internal Examiner)   **(External Examiner)**

CERTIFICATE

This is to certify that Mr. Adarsh G (20Z204), Mr. Elanthamil R (20Z215), Mr. Jeevan Krishna K V (20Z220), Mr. Nirmal M (20Z267) and Mr. Ajay Deepak P M (21Z431) of B.E. (CSE) semester 6 have been working on a project entitled "Effective Prediction of Parkinson's Disease" for Serene Technologies from January 2023 to April 2023 as part of the course on 19Z620 – Innovation Practices.

Place: Coimbatore
Date: 12-04-2023

Dr. Gopika Rani N,
Assistant Professor(SG),
Department of Computer Science and Engineering,
PSG College Of Technology,
Coimbatore – 641004

COUNTERSIGNED

HEAD
Department of Computer Science and Engineering,
PSG College Of Technology,
Coimbatore – 641004.

SERENE TECHNOLOGIES
Peace at IT's Best

Ref:

This is to certify that the project titled **"EFFECTIVE PREDICTION OF PARKINSON'S DISEASE"** for **Serene Technologies, Coimbatore,**was completed by the students studying 3rdyear B.E Computer Science & Engineering at PSG College of Technology, Coimbatore, under the guidance of **Dr.N.Gopikarani,** Assistant Professor (SG), Dept of CSE, PSG College of Technology, and the duration of the project is from Jan 2023 to April 2023.

The students involved in the projects are as given below:

1. 20Z204 ADARSH G
2. 20Z215 ELANTHAMIL R
3. 20Z220 JEEVAN KRISHNA K V
4. 20Z267 NIRMAL M
5. 21Z431 AJAY DEEPAK P M

Date: 11.04.2023

B.Vijay

Authorized Signature

(for Serene Technologies)

ACKNOWLEDGEMENT

We would like to express our sincere gratitude to **Dr. K. Prakasan**, Principal, PSG College of Technology, for giving us the opportunity to do a project with various facilities and infrastructure, without which the success of the project would not have been possible.

We extend our heartfelt thanks to **Dr. G. Sudha Sadasivam**, Professor and Head, Department of Computer Science and Engineering, PSG College of Technology, for her unfailing support throughout this project.

We express our sincere thanks to a Program Coordinator **Dr. N. Arul Anand**, Professor, Department of Computer Science and Engineering, PSG College of Technology, whose constant support and everlasting enthusiasm made it possible to be completed within the time.

We heartily thank our guide **Dr. N. Gopika Rani**, Assistant Professor(SG), Department of Computer Science and Engineering who was always there to help us and played a major role in the completion of the project and we wish to thank her for her enduring guidance.

We also wish to express our sincere thanks to **Ms. T. Anusha**, Assistant Professor(SG), Department of Computer Science and Engineering, for her guidance that played a vital role in completing my project on time.

Finally we would like to thank the faculty members, staff of CSE Department and friends without whom this project work would not have been completed successfully.

SYNOPSIS

Parkinson's disease is a progressive neurodegenerative disorder of the nervous system that affects the movement of human beings. The early detection and accurate prediction of Parkinson's disease can be challenging, but with the right tools and techniques, it is possible to identify those at risk and intervene early.

Due to the changes in speech patterns and vocal features that occur in patients with this condition, the voice is a potential biomarker for Parkinson's disease. Recent research has demonstrated that speech recordings may be analyzed to accurately discriminate between people with and without Parkinson's disease.

Voice recordings from people with and without Parkinson's disease will be collected for the study, and machine learning techniques will be used to analyze the data. The algorithms are able to recognise patterns and characteristics in the voice that are unique to Parkinson's disease. These patterns can then be utilized to determine a person's likelihood of developing Parkinson's disease.

The development of a non-invasive and accurate tool for predicting Parkinson's disease using voice could revolutionize the way the disease is diagnosed and treated. It might give medical professionals a Parkinson's disease early warning system, enabling earlier and more efficient treatment. Additionally, it could improve the standard of life for individuals with Parkinson's disease by enabling them to receive treatment sooner and manage their symptoms more effectively.

An improved performance of accuracy 94.87% is achieved by the prediction model for Parkinson’s Disease with the ensemble model compared to the 92.30% accuracy by the existing model.

CONTENTS

CHAPTER **Page No.**

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

INTRODUCTION

1.1 Overview

Parkinson's disease can be defined as a neurological disorder that affects the human nervous system. It is primarily due to the loss of neurons that produce dopamine in the human brain. Dopamine is a neurotransmitter (transmits chemical and electrical signals between neurons) that regulates movement, motivation, and reward in the human brain. The symptoms of Parkinson's disease develop gradually and include tremors, stiffness, slowed movements, and impaired balance and coordination. As the disease progresses, individuals may also experience cognitive and behavioral changes, such as dementia, depression, and anxiety. Parkinson's disease patients generally encompass low volume noise with a monotone quality and this method explores the classification of audio signals feature dataset to diagnose Parkinson's sickness, the classifiers that are to be defined using Machine Learning.

Parkinson’s disease can be characterized by tremors, rigidness and bradykinesia(slowness of movement) which causes the affected people difficulty in flexible movement, paucity of voluntary movement and irregularities in speech.These significant differences in features can be very well used to create a mathematical model to almost accurately diagnose Parkinson’s disease.

Features of voice like Average Vocal Voice Fundamental Frequency, Maximum and Minimum Vocal Fundamental Frequency and Jitter can be used in the mathematical model to predict the disease. Conventional algorithms like decision trees, Naive Bayes, Random forest predict the diagnosis of the disease with high variance. Hybrid models combining several high end mathematical models will help us to predict the disease more accurately.

The models which use unsophisticated conventional Machine Learning algorithms produce less accuracy. Thus, in this project the objective entails to design an enhanced machine learning model which could take data sets to predict the diagnosis of Parkinson’s Disease.

1.2 Motivation

The motivation for this paper is driven by the urgent need for early and accurate prediction of Parkinson's Disease (PD) to improve patient outcomes and disease management.

Machine learning (ML) models have shown great potential in leveraging diverse datasets to develop predictive models for PD, and this paper aims to provide a comprehensive review of the current state-of-the-art in ML-based PD prediction, addressing the challenges, opportunities, and future directions of this field.

1.3 Problem Statement

The problem of accurately and early predicting Parkinson's Disease (PD) remains a challenge, as traditional clinical assessments may not be sensitive enough to detect subtle markers of PD at an early stage. Existing prediction methods lack the precision, scalability, and interpretability needed for widespread clinical implementation. Therefore, there is a need to develop robust and interpretable machine learning (ML) models that can leverage diverse datasets to accurately predict PD, enabling early intervention strategies and improving patient outcomes.

1.4 Objective

The objective of this paper is to comprehensively review and analyze the current state-of-the-art in the development of machine learning (ML) models for prediction of Parkinson's Disease (PD) and provide an enhanced model for prediction of PD. This includes exploring various ML techniques, data types, and challenges associated with PD prediction.

The paper aims to provide insights into the strengths, limitations, and future directions of ML-based PD prediction, and highlight opportunities for improving accuracy, interpretability, and clinical implementation of ML models for PD prediction.

CHAPTER 2

LITERATURE SURVEY

2.1 Related Works

A Dinesh, et.al (2017) propose a predictive model that can effectively diagnose PD using a Boosted Decision Tree,which was an ensemble model made from gradient boosted regression trees and had a oscillating accuracy between 90-95%.It was also discovered that the strongest weighted features were spread1,spread2 and PPE. [1]

Muthumanickam S, et.al (2018) utilized several machine learning classifiers including Support Vector Machine (SVM), Feedforward BackPropagation Based Artificial Neural Network (FBANN), Random Tree (RT), Binary Logistic Regression, Linear Discriminant Analysis (LDA), Convolutional Neural Network (CNN), and Deep Belief Network (DBN) in their study. They found that Linear Regression was easy to comprehend and could be adjusted to prevent overfitting. Additionally, they used the SGD command to update linear models. In their experiments, they observed that Binary Logistic Regression produced interpretable algorithms and outputs, and achieved higher accuracy than deep neural networks.[2]

T J Wroge, et.al (2018) explores the effectiveness of using supervised classification algorithms such as deep neural networks,to accurately diagnose individuals with the disease.Their work which provided a peak accuracy of 85%.[3]

A U Haq, et.al (2019) proposes the use of L1 norm of feature selection of support vector machines can be used to select highly related features for the classification of PD and healthy person.The model used for their research was SVM.K-fold cross-validation where k=10 was applied to select the optimal values of tuning parameters.[4]

S Mohan, et.al (2019) worked on prediction of Heart Disease through Hybrid ML classification algorithms.Diverse data mining approaches and prediction methods, such as KNN, LR, SVM, NN, and Vote have been rather popular lately to identify and predict heart disease.They produces an enhanced performance level with an accuracy of 88.7%.[5]

Wu Wang, et.al (2020) employed Stochastic Gradient Descent (SGD) for training data models using a Feed-Forward Neural Network (FNN). They found that the linear discriminant analysis approach had the highest sensitivity, making it the most effective method for distinguishing between real patients. The proposed deep learning model achieved an accuracy rate of 96.45%, which was attributed to its ability to learn linear and nonlinear features from PD data without the need for manual feature extraction.[6]

K Mohan Rao, et.al (2020) throws a new light on prediction of parkinson's disease in which they use a drawing dataset for identifying patterns in people affected by parkinson's. Parkinson's has another predominant symptom rather than shrieks and shudder in voice which is tremors, muscle stiffness and slowness of movement.Their idea was to exploit this by making use of drawing data especially spiral drawing images which involves significant cognizance to draw one.[7]

S. Raval, et.al (2020) provides a comparative analysis of different machine learning models like Logistic Regression (LR), Support Vector Machine (SVM), Decision Tree (DT), K-nearest neighbors (KNN), Stochastic Gradient Descent (SGD) and Gaussian Naive Bayes (GNB) in prediction of parkinson's disease.Among all the tests, applying Random Forest (RF) on Static Spiral Test (for detecting tremor) gave the most significant result.[8]

Shrihari K Kulkarni, et.al (2021) utilized various machine learning algorithms including Decision Tree, Logistic Regression, Naive Bayes, and Deep Learning techniques such as Recurrent Neural Networks (RNN) to develop a model that predicts performance parameters. They also employed machine learning methods such as Logistic Regression, Random Forests, and Support Vector Machine for subject and record validation. The goal of their study was to differentiate early Parkinson's Disease from healthy individuals using the Movement Disorder Society-Unified Parkinson's Disease Rating Scale (MDSUPDRS).[9]

Yatharth Nakul, et.al (2021) compared various supervised learning algorithms, including Random Forest, Support Vector, and Naïve-Bayes, and evaluated their accuracy using a confusion matrix. They employed different classification methods and used ML classification techniques to improve accuracy and reduce potential errors. Additionally, hyperparameter tuning was performed to achieve the highest possible accuracy. The main goal of their study was to develop a model that could accurately classify data using machine learning techniques.[10]

CHAPTER 3

SYSTEM SPECIFICATIONS

3.1 Hardware Requirements

The minimum hardware requirements required for this project are

Feature	Specification
Graphics(GPU)	Radeon Pro 5500 M or Intel UHD 620
Processor	Dual core 2.0 GHz
RAM	4 GB
Storage	16 GB

Table 3.1: Hardware Requirements

3.2 Software Requirements

The minimum software requirements required for this project are

Feature	Specification
Operating System	Windows / Mac OS / Linux
Coding Platforms	VS Code / Google Colabs / Spyder / Jupyter Notebook
Programming Language	Python
Libraries	Numpy, Pandas, Matplotlib, Seaborn, Sklearn

Table 3.2: Software Requirements

3.3 Functional Requirements

- **Data collection and preprocessing**

The system should be able to collect data on assets, transactions and other relevant variables and preprocess the data for analysis. This could include data cleaning, transformation, and normalization.

● **Feature selection and engineering**

The system should be able to select or engineer features that are relevant to assets. This could include technical indicators, statistical measures(such as frequency), or other features that capture patterns or trends in the data.

● **Prediction and reporting**

The system should be able to make suggestions for future assets and generate reports or visualizations that communicate the suggestions to users.

● **Real-time processing**

The system should be able to handle real-time data streams and make suggestions in near real-time, if needed.

3.4 Non-Functional Requirements

3.4.1 Performance Requirements

● **Speed** **:** The system should be smooth and fast.

● **Accuracy** **:** The system should be very accurate and not make any errors.

3.4.2 Safety Requirements

● The system should guarantee the safety and security of the data so that it does not get corrupted, deleted, etc.

● The system should not crash due to the processing of the largest number of data.

3.4.3 Security Requirements

● The whole system is secured from outside access.

3.4.4 Software Quality Attributes

The quality attributes of the software are given below:

● **Adaptability** **:** This system should be adaptable by any organization.

● **Correctness** **:** The results of the function must be accurate.

● **Maintainability** **:** After the deployment of the project if any error occurs then it must be easily maintained by the software developer.

● **Portability** **:** The system must be deployable at any machine that has sufficient processing power and memory to run the project.

● **Reliability** **:** The performance of the system must be better, which increases the reliability of the software.

● **Reusability** **:** The data and records are saved in the database and can be reused if needed.

- **Robustness** **:** If there is any error in any window or module then it should not affect the remaining part of the system.
- **Testability** **:** The system is tested at every stage.
- **Usability** **:** The system is easy to use by a wide range of users.
- **Productivity** **:** The system will produce every desired result accurately.
- **Cost effective** **:** This system is cost efficient and is bearable by any organization.

CHAPTER 4

SYSTEM DESIGN

4.1 Existing Method

Existing methods involve usage of deep learning techniques which are computationally costly and time consuming using static spiral diagram datasets and voice datasets.

Some of the drawbacks of existing models are:

- The regulation of data collection techniques is weak, which can lead to unreliable results such as out-of-range or non-existent data. Some existing models solely rely on the evaluation of movements, while there are other sources of data available on both PD-affected and healthy individuals.
- Deep Learning models gave delayed results and had slow output generation and the best proposed methodology gives higher error rate when confusion matrix is plotted.
- Deep Learning like neural networks is a black box algorithm which is difficult to evaluate. Deep learning algorithms are difficult to comprehend and understand due to the underlying complexity.
- Some models used linear regression models which do not provide a reliable means to model non-linear relationships.

The accuracy of existing models are also low when compared using confusion matrices.

4.2 Proposed System

The proposed system consists of two main components or modules:

- Data Collection and Feature Extraction Module

This module is responsible for collection of data and extracting necessary features that impact the efficiency of the model.The Module must then store the collected data into a reliable database from which the model can access or retrieve data easily.

- Prediction Module

This module contains the hybrid machine learning model which is a data specific model developed to effectively predict the diagnosis of parkinson's disease.

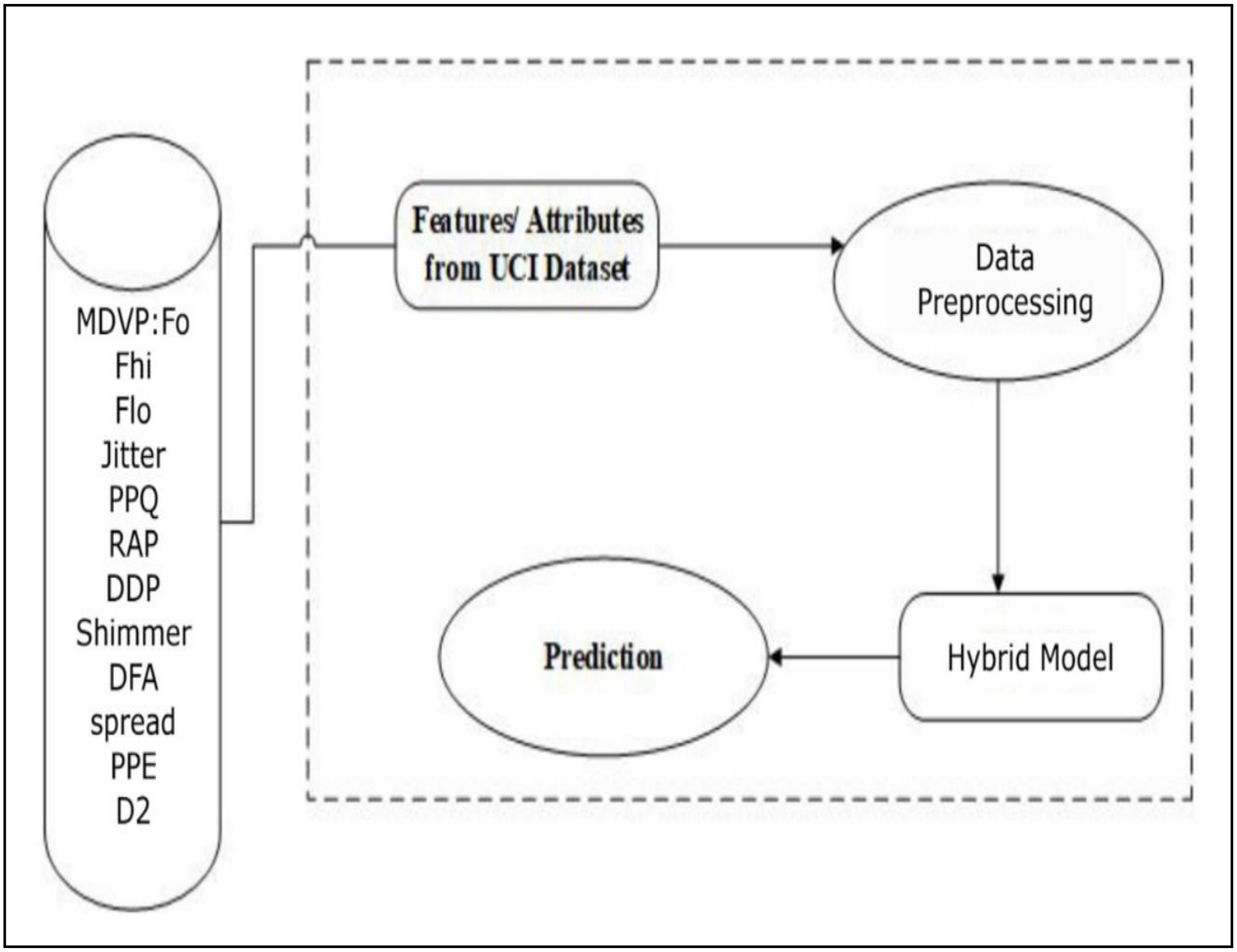


Fig. 4.1: System Design

4.2.1 System Architecture

The parkinson’s disease prediction system collects data from audio or voice recording samples and uses a hybrid machine learning algorithm to effectively predict the disease.The software includes data extraction,model training and testing modules and finally a prediction module.The training and testing data followed by prediction data are stored in database so the model can access them easily and efficiently. The system predicts the probability of a person being affected or diagnosed with parkinsons as output.

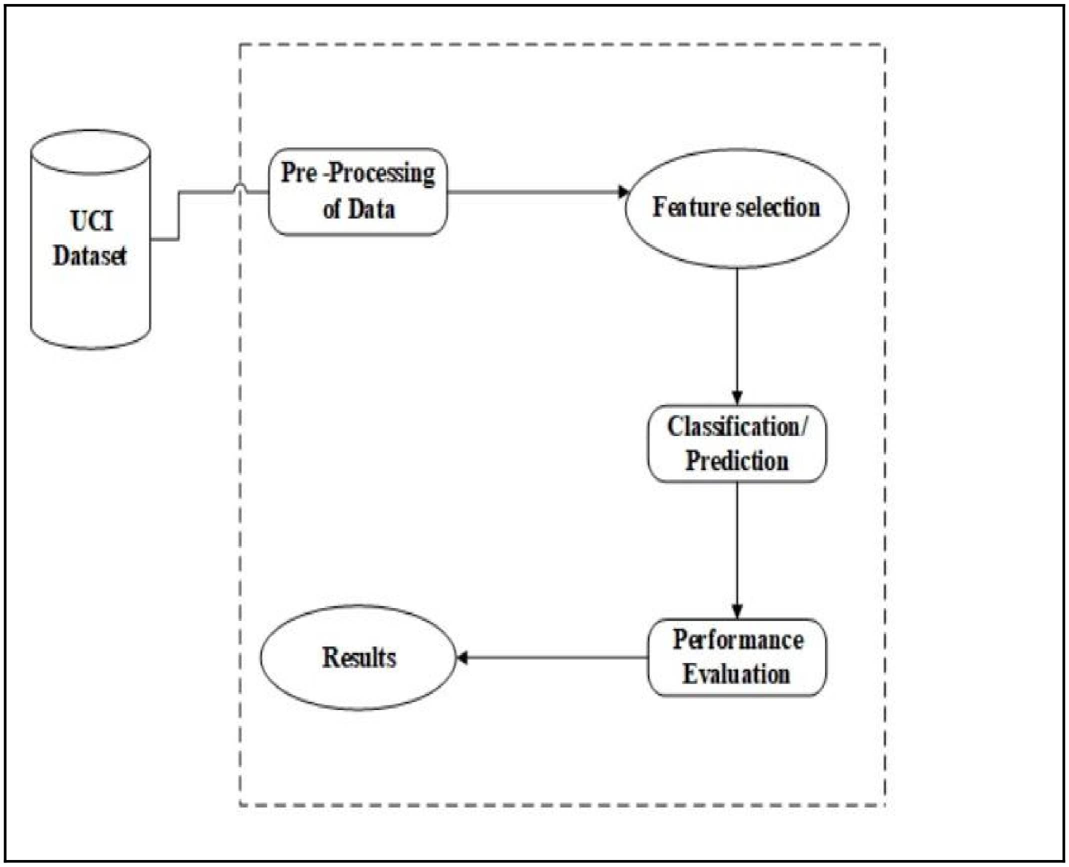


Fig. 4.2: System Architecture

4.2.2 Various Steps involved in Prediction

- Data Collection : Collection of relevant data from individuals regarding Parkinson's disease.
- Data Preprocessing : Cleaning, transforming, and normalizing the collected data to prepare it for analysis.
- Feature Engineering : Creation of new features from the collected data to improve the accuracy of the analysis.
- Exploratory Data Analysis : Initial analysis of the collected data to identify patterns and relationships.
- Model Selection : Selection of appropriate machine learning models for the analysis of Parkinson's disease.
- Model Training : Training of the selected machine learning models using the collected data.
- Model Evaluation : Evaluation of the performance of the trained models using various metrics such as accuracy, sensitivity and specificity.

- Model Comparison : Comparison of the performance of the different machine learning models to determine the best model for the analysis of Parkinson's disease.
- Model Fine-Tuning : Fine-tuning of the best model to improve its performance.
- Model Interpretation : Interpretation of the results of the analysis to understand the relationships between the features and Parkinson's disease.
- Visualization: Creation of visual representations of the results to aid in the interpretation and understanding of the analysis.
- Clinical Validation : Validation of the results of the analysis with healthcare professionals to ensure their accuracy and usefulness in the management of Parkinson's disease.

4.2.3 Ensemble

Ensemble methods are useful in situations where a single model may not be able to capture all the patterns in the data, and where combining multiple models can lead to better predictive performance. However, ensembling can also increase the computational cost and complexity of the modeling process, and requires careful tuning of hyperparameters to achieve optimal performance.

The Algorithm is explained as below

```
Input:
- Dataset: training_set, validation_set, test_set
- Number of Base Models: num_base_models
- Maximum Depth of Decision Trees: max_depth
- Number of Data Points in Each Subset: subset_size

Output:
- Ensemble Prediction: ensemble_prediction

# Train Base Models
for i = 1 to num_base_models:
    subset = randomly_select(training_set, subset_size) # Randomly select subset_size data points from training_set
    base_model = train_decision_tree(subset, max_depth) # Train a decision tree on the subset with max_depth as the maximum depth
    store base_model as base_model_i

# Combine Base Models
ensemble_prediction = []
```

```
for each data_point in test_set:
    base_model_predictions = []
    for i = 1 to num_base_models:
        prediction = predict(base_model_i, data_point) # Get prediction from base_model_i for the data_point
        append prediction to base_model_predictions
    ensemble_prediction.append(majority_vote(base_model_predictions)) # Use majority voting to combine predictions

# Evaluate Ensemble
ensemble_accuracy = evaluate(ensemble_prediction, test_set)

# Return Ensemble Prediction
return ensemble_prediction, ensemble_accuracy
```

Table 4.1: Ensemble Algorithm

The input for the ensemble algorithm includes a dataset that is split into training, validation, and test sets, the number of base models to be trained, the maximum depth of decision trees, and the number of data points in each subset. The algorithm then trains the specified number of base models by randomly selecting a subset of the training set and training a decision tree on that subset with the maximum depth specified. The base models are then combined using majority voting to make predictions on the test set, and the ensemble prediction is evaluated for accuracy using the test set. Finally, the algorithm returns the ensemble prediction and accuracy. This ensemble algorithm can be used to improve the accuracy of machine learning models by combining multiple models trained on different subsets of the training data.

4.2.4 Hyperparameter Tuning

Hyperparameter tuning is an important step in machine learning, as the choice of hyperparameters can have a significant impact on the performance of the model. Grid search is the commonly used technique for hyperparameter tuning.

The Algorithm is explained as below

```
Input:
- Base Model: model
- Hyperparameter Grid: hyperparameter_grid
- Dataset: training_set, validation_set
- Evaluation Metric: evaluation_metric

Output:
- Best Hyperparameters: best_hyperparameters
- Best Model: best_model

# Initialize best hyperparameters and best model
best_hyperparameters = None
best_model = None
best_evaluation_score = -inf

# Loop over hyperparameter grid
for each combination of hyperparameters in hyperparameter_grid:
    # Train model with current hyperparameters
    model = train_model(training_set, model, hyperparameters)

    # Evaluate model on validation set
    predictions = predict(model, validation_set)
    evaluation_score = evaluate(predictions, validation_set, evaluation_metric)

    # Check if current model is better than previous best model
    if evaluation_score > best_evaluation_score:
        best_evaluation_score = evaluation_score
        best_hyperparameters = hyperparameters
        best_model = model

# Return best hyperparameters and best model
return best_hyperparameters, best_model
```

Table 4.2: Hyperparameter Tuning Algorithm

The input for this algorithm includes a base model, a hyperparameter grid, training and validation sets, and an evaluation metric. The algorithm loops over the hyperparameter grid and trains a model with each combination of hyperparameters on the training set. The model is then evaluated on the validation set using the specified evaluation metric. If the evaluation score of the current model is better than the previous best model, then the current model becomes the new best model. Finally, the algorithm returns the best hyperparameters and best model based on the evaluation metric. This algorithm can be used for hyperparameter tuning of a given base model to find the optimal set of hyperparameters that yield the best performance on the validation set.

4.2.5 Feature Importance

Feature importance is a powerful technique for feature selection and can be used to identify the most important features in a dataset. However, feature importance scores can be affected by the presence of correlated features and may not always provide a complete picture of the relationship between features and the target variable.

The Algorithm is explained as below

```
Input:
- Dataset: data
- Machine Learning Model: model

Output:
- Feature Importance: feature_importance

# Step 1: Split the dataset into training and testing sets
train_data, test_data = train_test_split(data)

# Step 2: Train the machine learning model on the training set
train_model = train_model(train_data, model)

# Step 3: Calculate feature importance using the trained model
feature_importance = calculate_feature_importance(train_model, train_data)

# Step 4: Sort feature importance in descending order
sort feature_importance in descending order by importance

# Step 5: Select the top k important features
k = determine_k() # Determine the value of k based on domain knowledge or through
cross-validation
selected_features = select_top_k_features(feature_importance, k)

# Step 6: Extract the selected features from the dataset
train_data_selected = extract_features(train_data, selected_features)
test_data_selected = extract_features(test_data, selected_features)

# Step 7: Retrain the model on the selected features
train_model_selected = train_model(train_data_selected, model)

# Step 8: Evaluate the model on the testing set with selected features
predictions = predict(train_model_selected, test_data_selected)
evaluation_score = evaluate(predictions, test_data_selected)

# Return the selected feature importance and evaluation score
return selected_features, evaluation_score
```

Table 4.3:Feature Importance Algorithm

The code snippet is a workflow for feature selection in machine learning, which involves identifying the most important features in a dataset and training a model only on those selected features. Firstly, the dataset is split into training and testing sets using the train_test_split function. Then, the machine learning model is trained on the training set using the train_model function. Next, the feature importance is calculated using the calculate_feature_importance function, and then sorted in descending order. The top k important features are selected using the select_top_k_features function based on domain knowledge or cross-validation. The selected features are extracted from both the training and testing sets, and the model is retrained using only those selected features. Finally, the model's performance is evaluated on the testing set using the predict and evaluate functions, and the selected features and evaluation score are returned. This workflow helps to improve the performance of the model by reducing the dimensionality of the data and focusing only on the most important features.

CHAPTER 5

IMPLEMENTATION

The requirement for this project is analyzed from various resources and the dataset is obtained based on the requirements that are collected. The dataset is then preprocessed according to the requirements of traditional machine learning algorithms. Next, the preprocessed dataset is trained and tested on these algorithms. The conventional ML algorithms have less accuracy for the given dataset, therefore the goal is to propose a dataset-specific method that provides better accuracy than the traditional algorithms. Steps to increase the accuracy include finding important features, hyperparameter tuning and ensemble of two or more models. The dataset consists of 24 columns and 196 rows of voice data in csv format.

5.1 Data Preparation

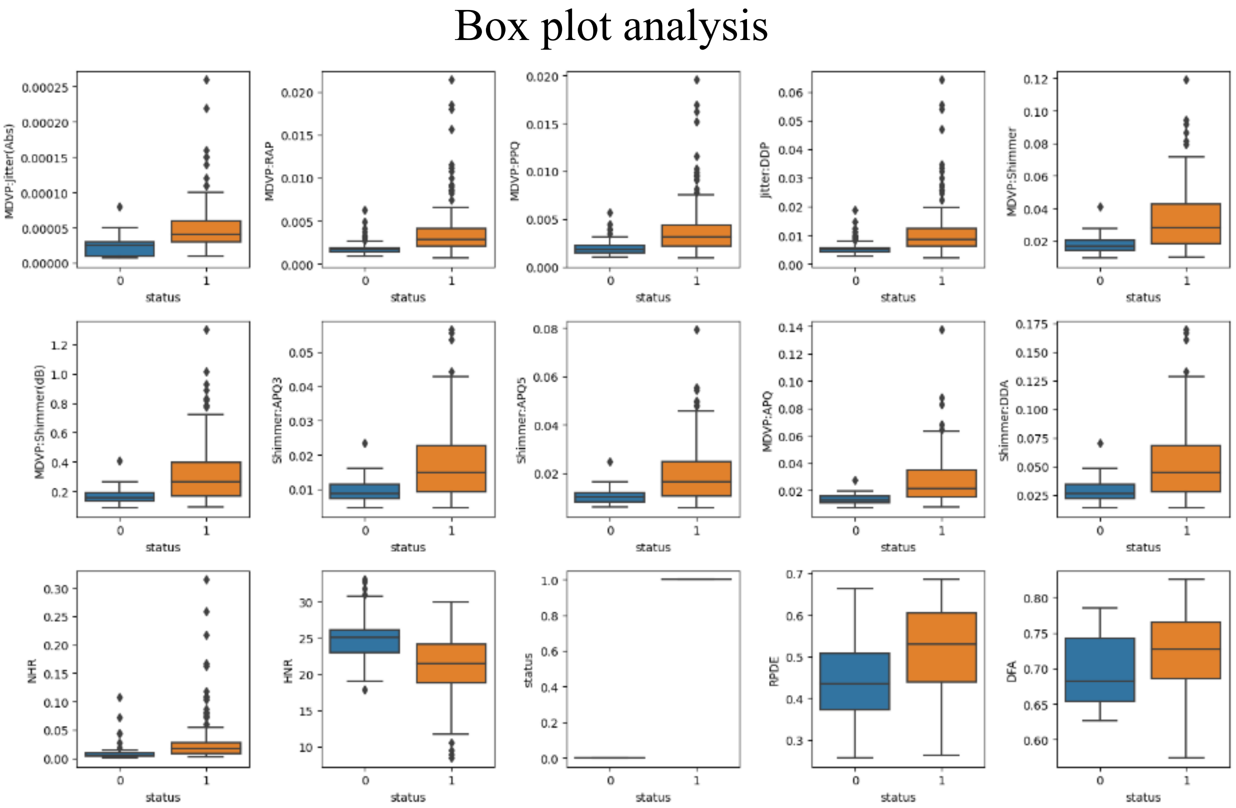


Fig. 5.1: Box Plot Analysis

Correlation

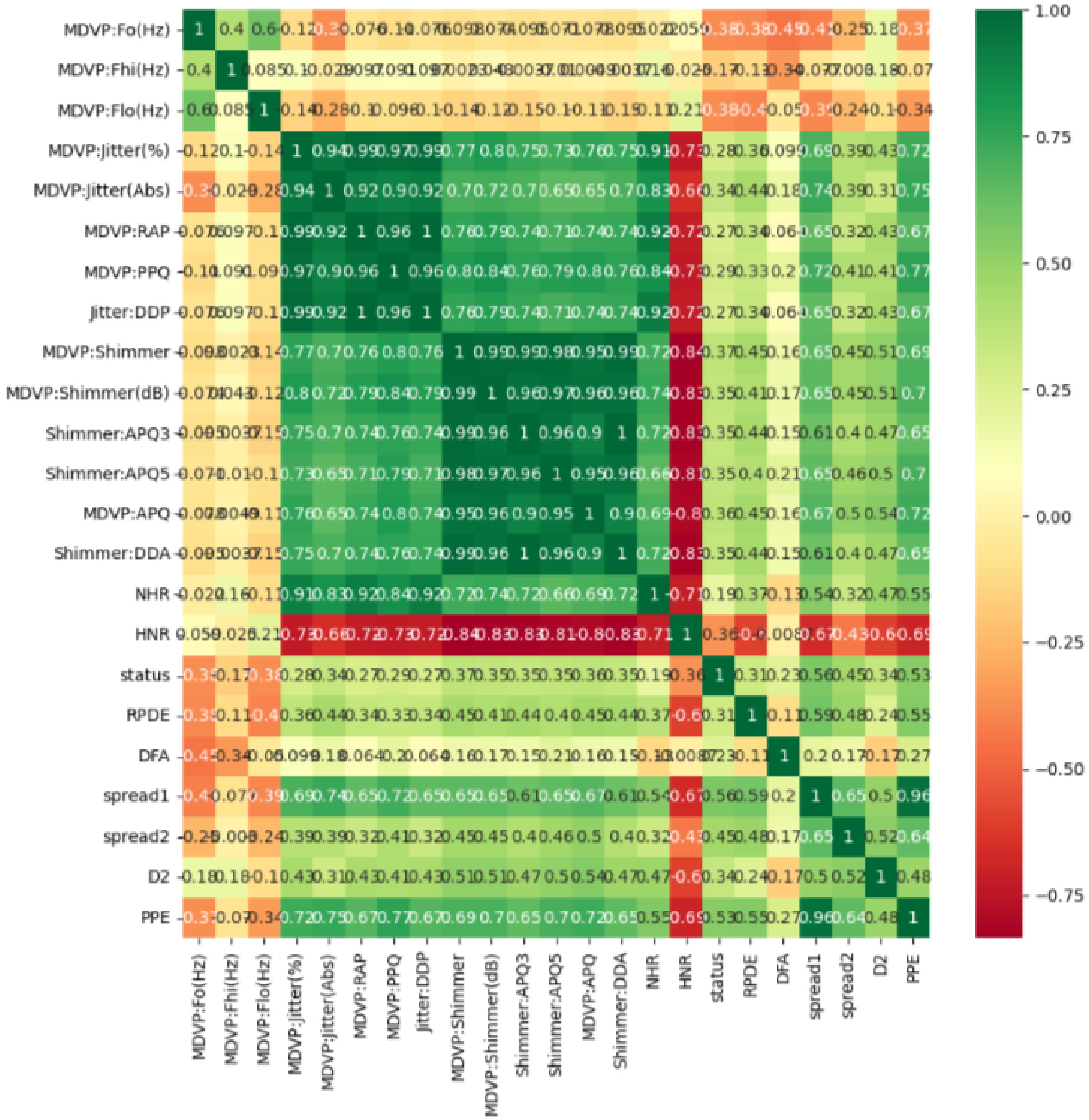


Fig. 5.2: Correlation matrix in heat map

Feature Scaling

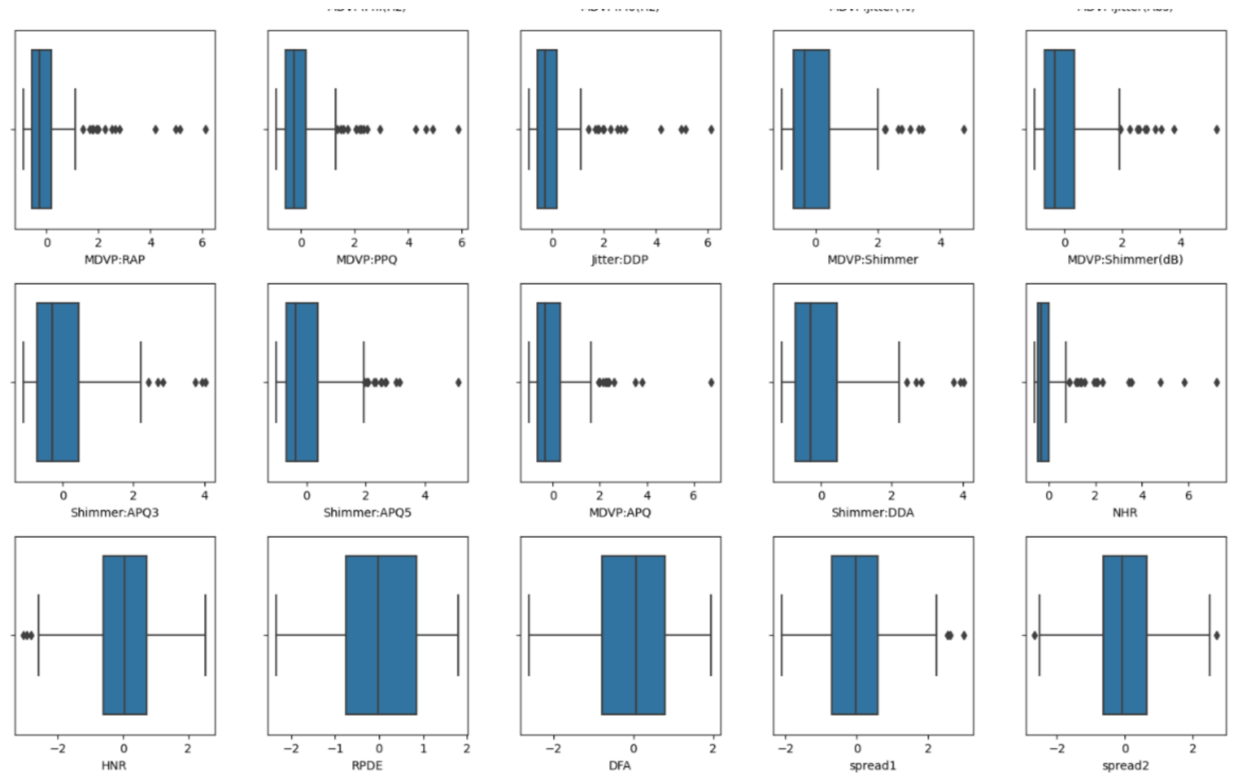
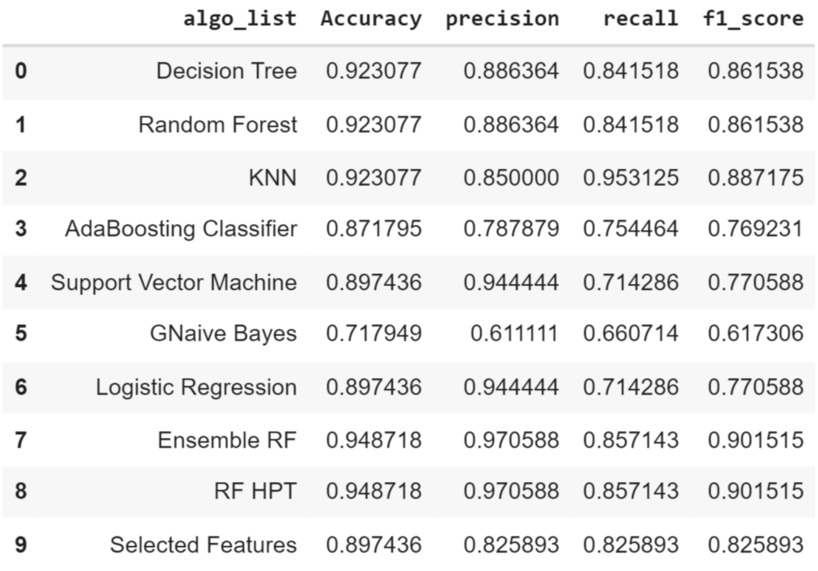


Fig. 5.3: Feature Scaling

5.2 Comparative Analysis

The comparative analysis between traditional machine learning algorithms and the proposed method is given in figure 5.4.



	algo_list	Accuracy	precision	recall	f1_score
0	Decision Tree	0.923077	0.886364	0.841518	0.861538
1	Random Forest	0.923077	0.886364	0.841518	0.861538
2	KNN	0.923077	0.850000	0.953125	0.887175
3	AdaBoosting Classifier	0.871795	0.787879	0.754464	0.769231
4	Support Vector Machine	0.897436	0.944444	0.714286	0.770588
5	GNaive Bayes	0.717949	0.611111	0.660714	0.617306
6	Logistic Regression	0.897436	0.944444	0.714286	0.770588
7	Ensemble RF	0.948718	0.970588	0.857143	0.901515
8	RF HPT	0.948718	0.970588	0.857143	0.901515
9	Selected Features	0.897436	0.825893	0.825893	0.825893

Fig. 5.4: Comparative Analysis Report

Accuracy

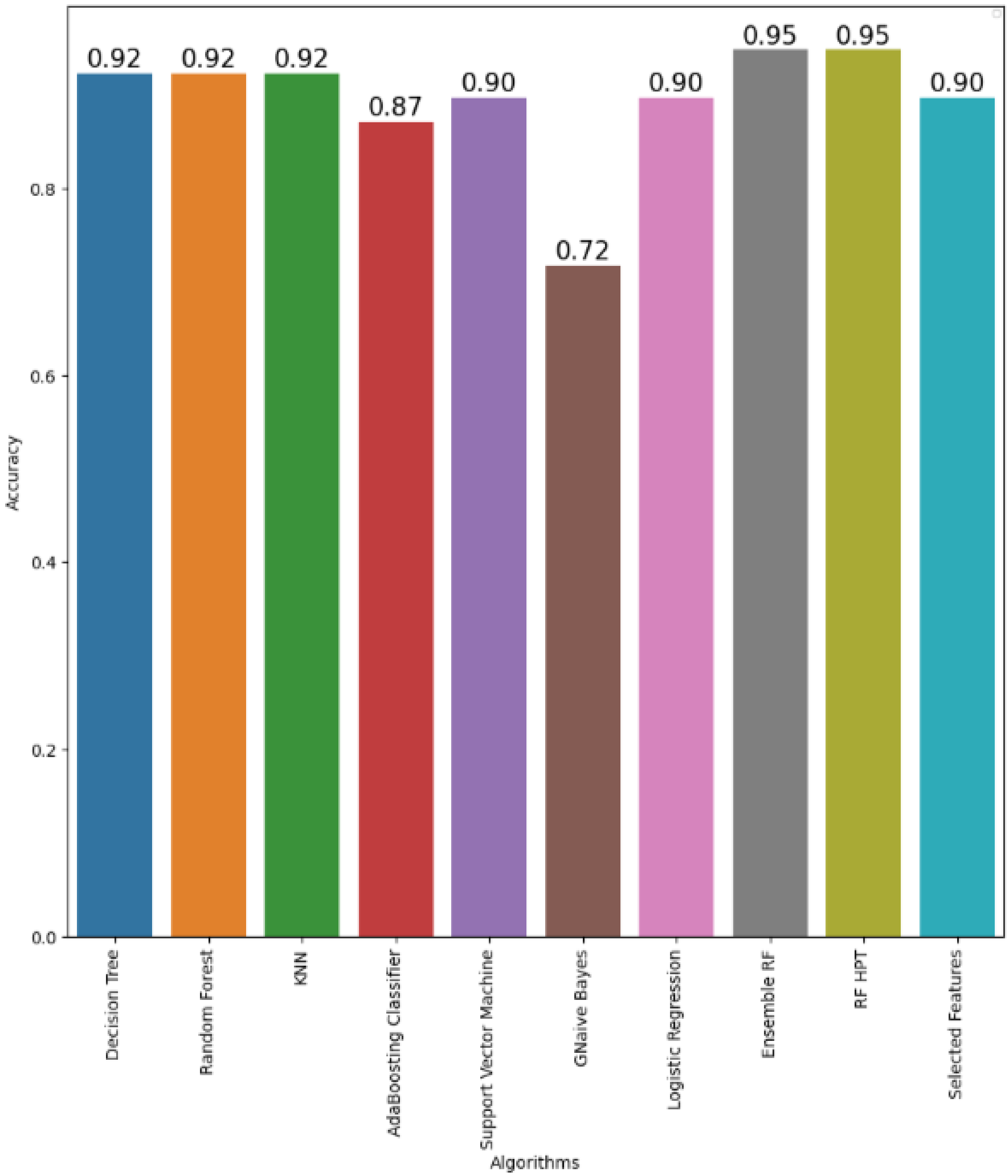


Fig. 5.5: Accuracy Report

The Fig. 5.5 provides a comparative analysis of the accuracies of existing models and ensemble model.The maximum accuracy provided by the existing model is 92.30% by random forest classifier and the ensemble model provides a greater accuracy of 94.87%.

Precision

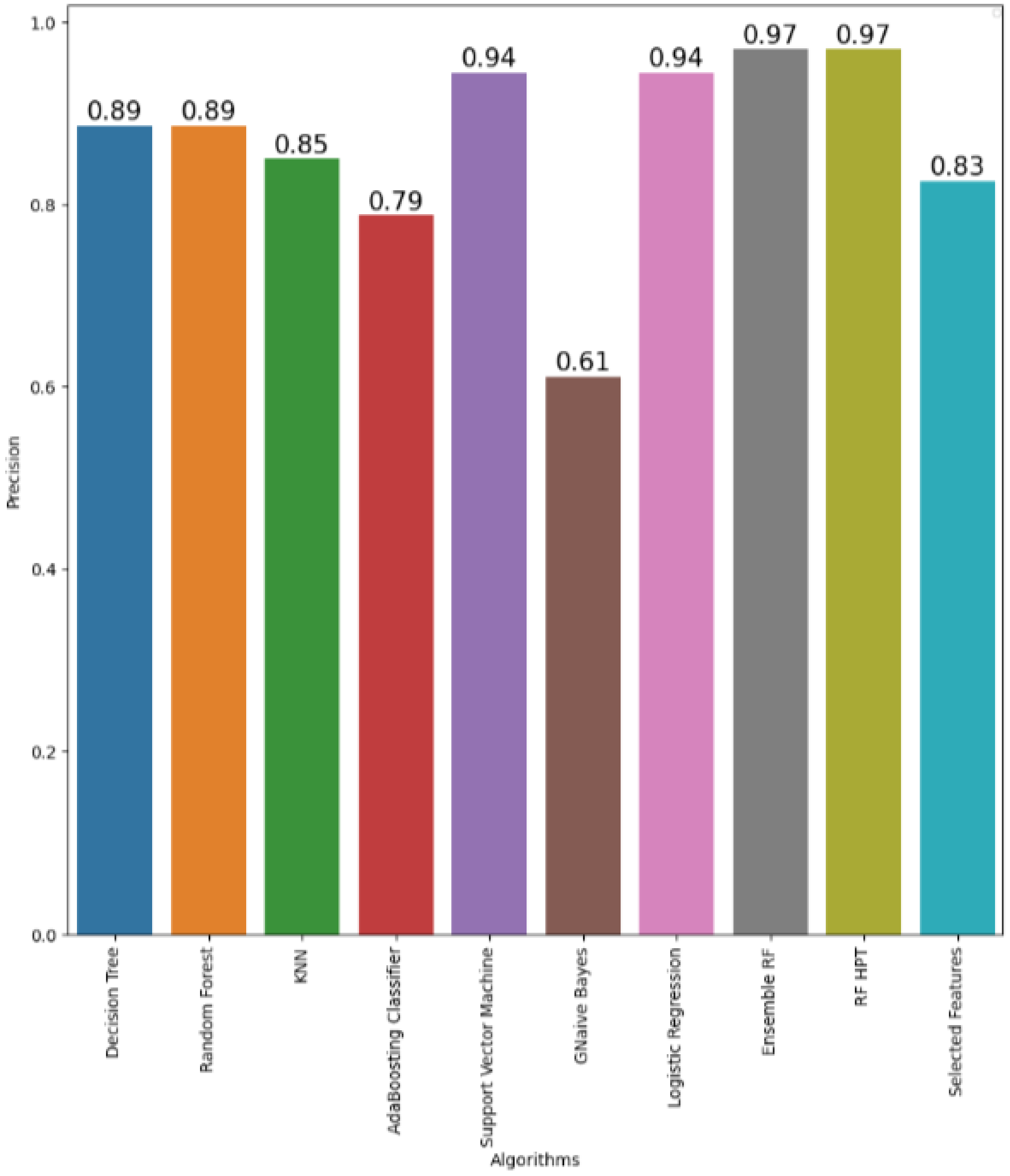


Fig. 5.6: Precision Report

The Fig. 5.6 provides a comparative analysis of the precisions of existing models and ensemble model. The maximum precision provided by the existing model is 94.44% by SVM classifiers and the ensemble model provides a greater accuracy of 97.05%.

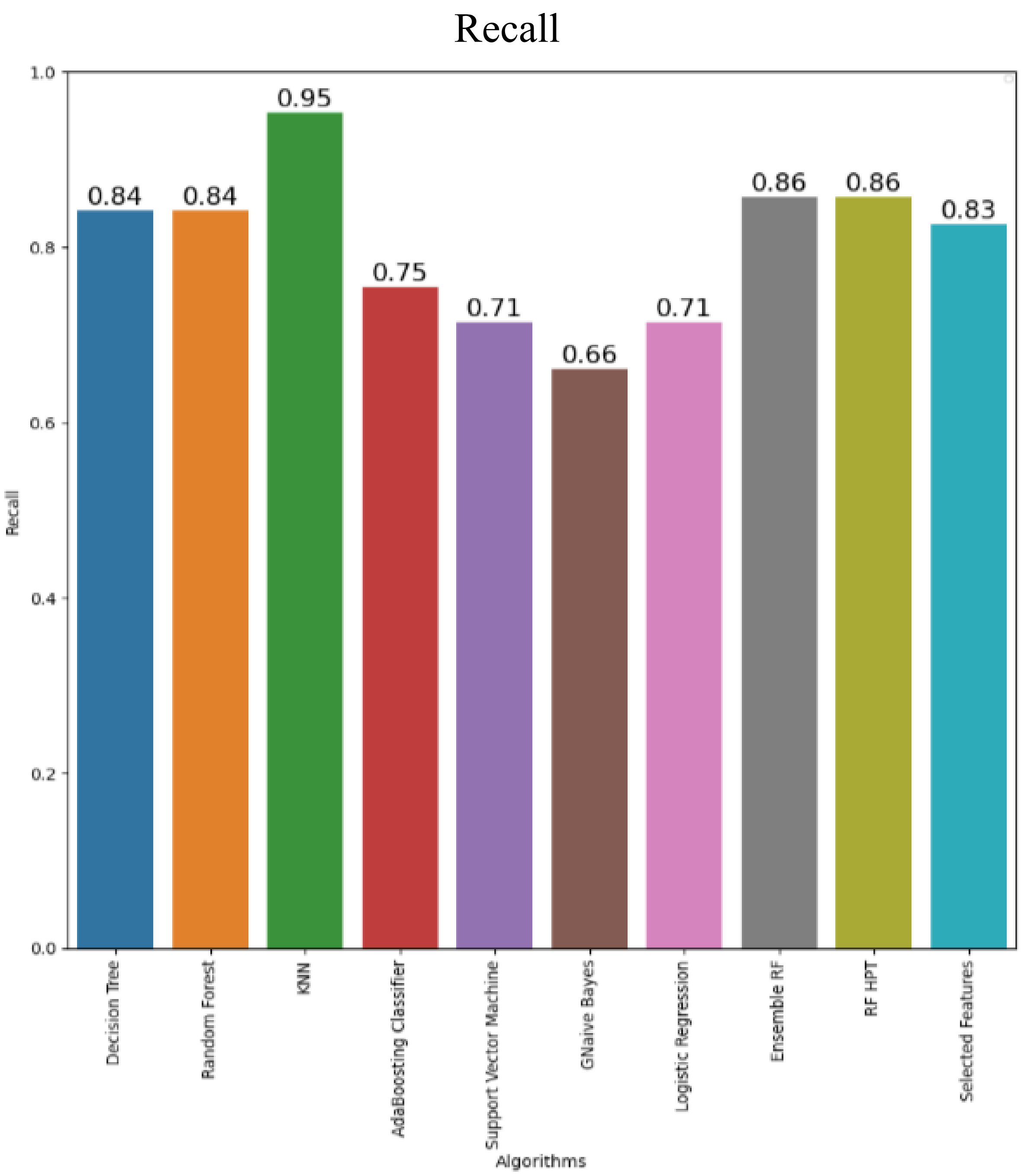


Fig. 5.7: Recall Report

The Fig. 5.7 provides a comparative analysis of the recalls of existing models and ensemble model. The maximum recall provided by the existing model is 95.31% by KNN and the ensemble model provides a recall of 86%. This is because KNN simply stores training instances in memory and makes predictions based on the similarity of test instances to the training instances.

F1 Score

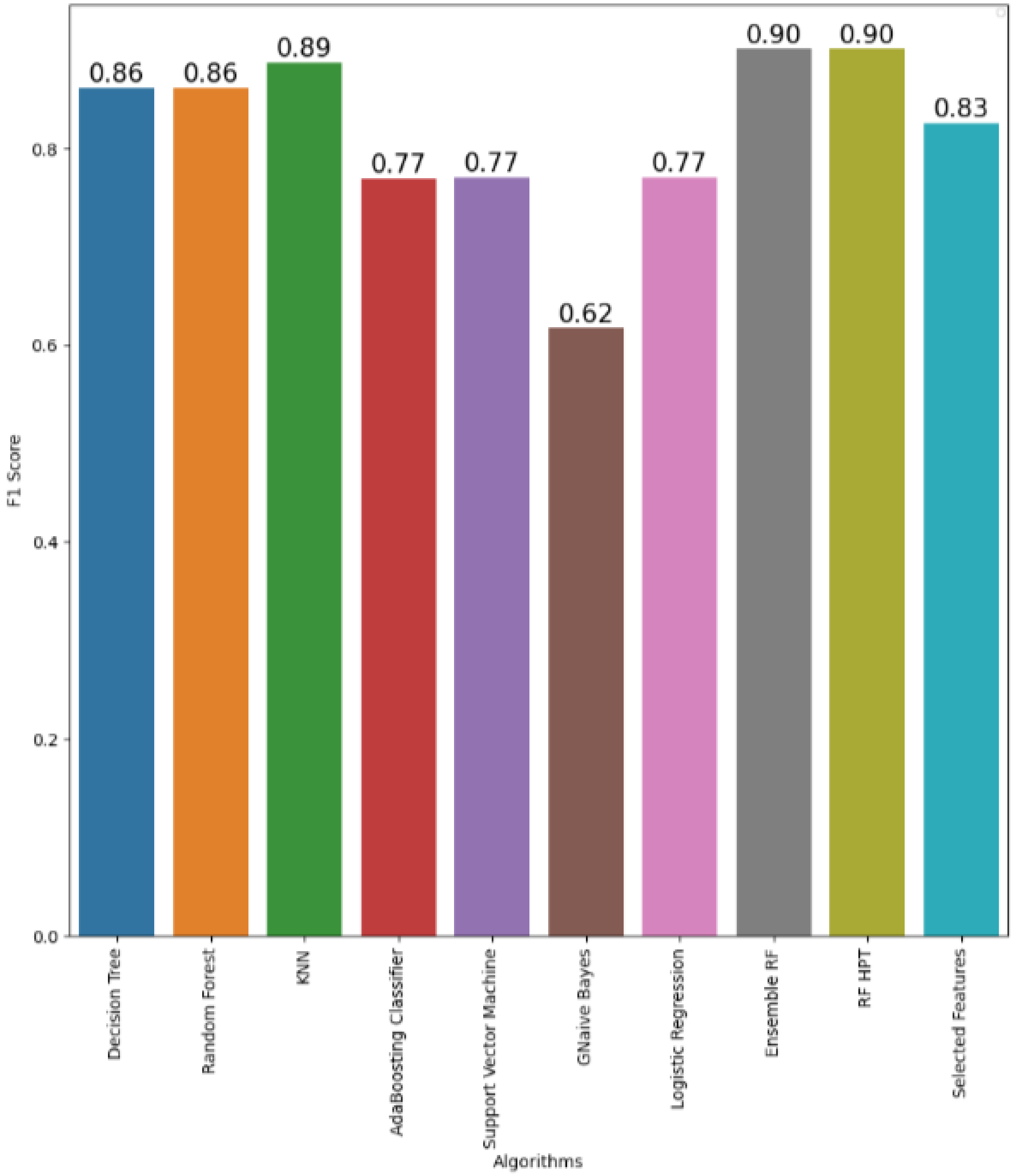


Fig. 5.8: F1 Score Report

The Fig. 5.8 provides a comparative analysis of the recalls of existing models and ensemble model. The maximum f1 score provided by the existing model is 88.71% for KNN classifiers and the ensemble model provides a greater f1 score of 90.15%.

Extraction of Important Features

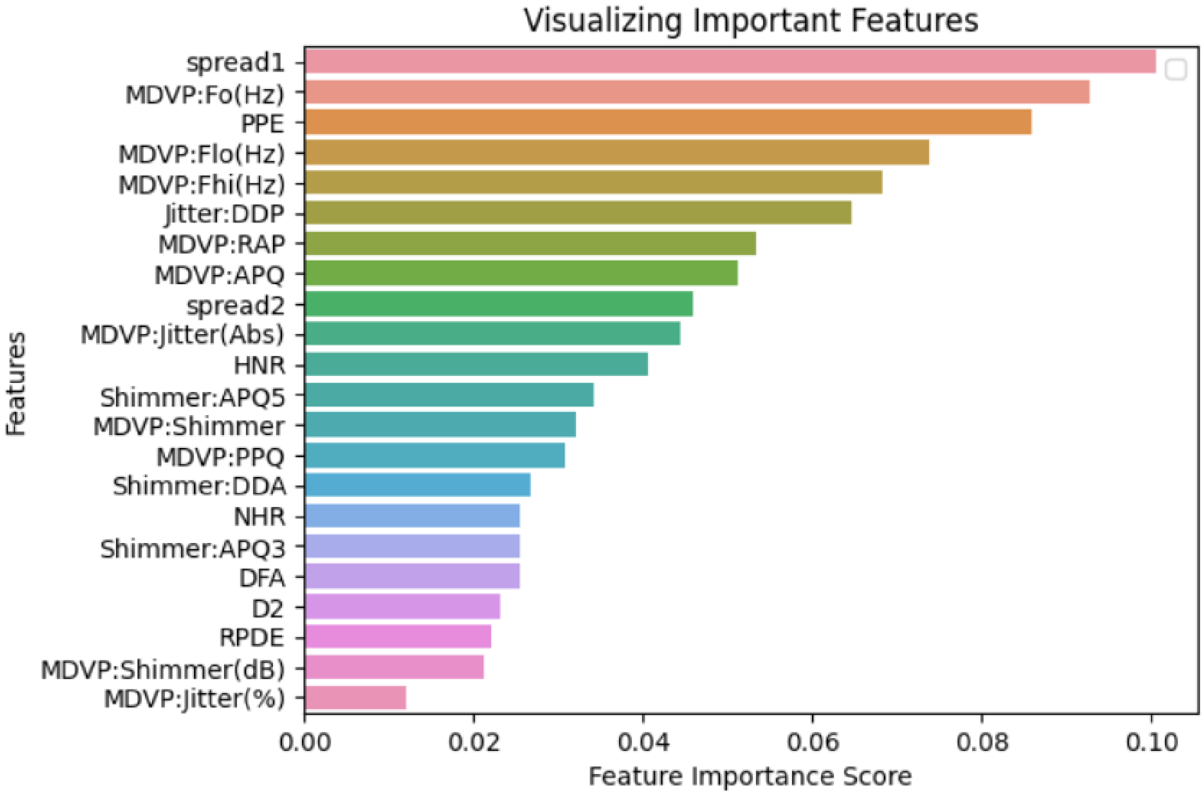


Fig. 5.9: Extraction of Important Features

To conclude, the ensemble model of Random Forests has demonstrated exceptional performance with an accuracy of 94.87%. This indicates that the model is capable of making accurate predictions on the dataset used for training and testing. The Random Forests ensemble technique, which combines multiple decision tree models, has proven to be a powerful approach for handling complex data and achieving high accuracy levels.

CHAPTER 6

CONCLUSION AND FUTURE ENHANCEMENTS

In conclusion, detecting Parkinson's disease using machine learning is a promising approach that can help improve early detection and personalized diagnosis, reduce the need for invasive procedures, and provide objective assessment of symptoms. By analyzing patient data, such as speech, gait, and motor symptoms, machine learning algorithms can identify patterns that are difficult for human experts to detect, leading to improved accuracy and efficiency in detecting Parkinson's disease.

As the technology continues to advance, machine learning has the potential to revolutionize Parkinson's disease diagnosis and treatment, improving the quality of life for patients living with this condition. Therefore, continued research and development in this area is crucial to improving Parkinson's disease management and patient outcomes.

The system is expected to undergo continuous enhancements in the future. These may include improved performance through optimized cloud-based resources and advanced caching mechanisms for the faster code compilation and producing output. Therefore, an efficient model for the effective prediction of Parkinson’s disease is developed.

The future enhancements can be performed by using multiple machine learning techniques for better prediction. Furthermore, different ensemble models by combining different models could be developed to give better performance.

REFERENCES

1. A. Dinesh and J. He, "**Using machine learning to diagnose Parkinson's disease from voice recordings**," 2017 IEEE MIT Undergraduate Research Technology Conference (URTC), Cambridge, MA, USA, 2017, pp. 1-4, doi: 10.1109/URTC.2017.8284216.

2. Muthumanickam S1 , Gayathri J2 , Eunice Daphne J3 ,, **"Parkinson's Disease Detection And Classification Using Machine Learning And Deep Learning Algorithms– A Survey"**, International Journal of Engineering Science Invention (IJESI) ISSN (Online): 2319 – 6734, ISSN (Print): 2319 – 6726 www.ijesi.org ,Volume 7 Issue 5 Ver. 1, May 2018 || PP 56-63.

3. T. J. Wroge, Y. Özkanca, C. Demiroglu, D. Si, D. C. Atkins and R. H. Ghomi, "**Parkinson's Disease Diagnosis Using Machine Learning and Voice**," 2018 IEEE Signal Processing in Medicine and Biology Symposium (SPMB), Philadelphia, PA, USA, 2018, pp. 1-7, doi: 10.1109/SPMB.2018.8615607.

4. A. U. Haq et al., "**Feature Selection Based on L1-Norm Support Vector Machine and Effective Recognition System for Parkinson's Disease Using Voice Recordings**," in IEEE Access, vol. 7, pp. 37718-37734, 2019, doi: 10.1109/ACCESS.2019.2906350.

5. S. Mohan, C. Thirumalai and G. Srivastava, "**Effective Heart Disease Prediction Using Hybrid Machine Learning Techniques**," in IEEE Access, vol. 7, pp. 81542-81554, 2019, doi: 10.1109/ACCESS.2019.2923707.

6. Wu Wang1 , Junho Lee2 , Fouzi harrou3 and Ying sun4,,,**" Early Detection of Parkinson's Disease Using Deep Learning and Machine Learning"** IEEE ACCESS Digital Object Identifier 10.1109/ACCESS.2020.3016062 Volume 8,2020-Page no 147635- 147646.

7. K. M. M. Rao, M. S. N. Reddy, V. R. Teja, P. Krishnan, D. J. Aravindar and M. Sambath, "**Parkinson's Disease Detection Using Voice and Spiral Drawing Dataset**," 2020 Third International Conference on Smart Systems and Inventive Technology (ICCSIT), Tirunelveli, India, 2020, pp.787-791, doi: 10.1109/ICSSIT48917.2020.9214276.

8. S. Raval, R. Balar and V. Patel, "**A Comparative Study of Early Detection of Parkinson's Disease using Machine Learning Techniques**," 2020 4th

International Conference on Trends in Electronics and Informatics (ICOEI)(48184), Tirunelveli, India, 2020, pp. 509-516, doi: 10.1109/ICOEI48184.2020.9142956.

9. Shrihari K Kulkarni1, K R Sumana2,**"Detection of Parkinson's Disease Using Machine Learning and Deep Learning Algorithms"** International Journal of Engineering Science Invention (IJESI) ISSN (Online): 2319 – 6734, ISSN (Print): 2319 – 6726 VOLUME 8 ISSUE:8, (AUG 2021)-Page No :1189-1192

10. Yatharth Nakul1 , Ankit Gupta2 , Hritik Sachdeva3,,," **Parkinson Disease Detection Using Machine Learning Algorithms**" International Journal of Science and Research (IJSR) ISSN: 2319-7064 SJIF (2020): 7.803 Volume 10 Issue 6, June 2021-Page no 314-318.